3. Loss Function

Standard loss/cost/objective function measures the error between y and the true value t.

- Sum of Squares for Error (SSE) $= \sum_{n=1}^{N}(t^{(n)} - y^{(n)})^2$
- Mean Squared Error (MSE) $= \frac{1}{N}\sum_{n=1}^{N}[t^{(n)} - y^{(n)}]^2$
- Root Mean Squared Error (RMSE) $= \sqrt{\frac{1}{N}\sum_{n=1}^{N}[t^{(n)} - y^{(n)}]^2}$
- Relative Squared Error (RSE) $= \frac{\sum_{n=1}^{N}[t^{(n)} - y^{(n)}]^2}{\sum_{n=1}^{N}[t^{(n)} - \bar{t}]^2}, \bar{t} = \frac{1}{N}\sum_{n=1}^{N} t^{(n)}$
- Mean Absolute Error (MAE) $= \frac{1}{N}\sum_{n=1}^{N}|t^{(n)} - y^{(n)}|$
- Relative Absolute Error (RAE) $= \frac{\sum_{n=1}^{N}|t^{(n)} - y^{(n)}|}{\sum_{n=1}^{N}|t^{(n)} - \bar{t}|}, \bar{t} = \frac{1}{N}\sum_{n=1}^{N} t^{(n)}$

4. Optimization

- Preprocess: incorporate the bias w_0 into **w** by using $x_0 = 1$ (Add an **1** to input **x**) . Then, **x** = $\begin{bmatrix} 1 \\ x \end{bmatrix}$
- Linear regression model: $y(x) = w^T\mathbf{x}$
- MSE loss: $l(w) = \frac{1}{2N}\sum_{n=1}^{N}[t^{(n)} - y(x^{(n)})]^2$, convex

4.1 Least square solution

i. let the gredient equal to 0, to find the minima: $\nabla l(w) = -\frac{1}{N}\sum_{n=1}^{N}(t^{(n)} - w^T\mathbf{x}(n))\mathbf{x}^{(n)}$ = 0

ii. then we get: $w = (\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T t$

4.2 Gradient decent

- let gradient decrease to the smallest through iteration: Initalize at one point, calculate its gradient and move in the opposite direction.
- Protocol:
  a. initialize w (randomly)
  b. repeatedly update w based on the gradient, λ is the learning rate

7. Normalization

- Improving model accuracy: Comparability in values between features across different dimensions can significantly enhance the accuracy of model learning.
- Accelerating learning convergence: Searching for the optimum becomes notably smoother, making it easier for the model to converge correctly to the optimal solution.

7.1 Min-Max normalization

$$x^* = \frac{x - x_{min}}{x_{max} - x_{min}}$$

- Maps the data along any dimension to [0,1]
- The purpose of **min-max normalization**: make the impact of each feature compatible, which involves scaling transformations of the features
- Normalizing data will alter the distribution of the feature data.

7.2 Mean normalization

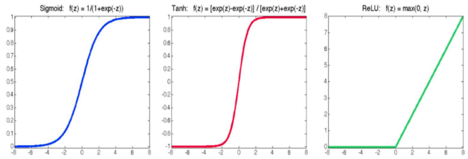
$$x^* = \frac{x - \mu}{\sigma}$$

where

$$\mu = \frac{1}{N}\sum_{n=1}^{N} x^{(i)}, \sigma^2 = \frac{1}{N}\sum_{n=1}^{N}(x^{(i)} - \mu)^2$$

- The data becomes **zero mean** and **unit variance**
- **Mean normalization**: aims to make different features comparable to each other
- The distribution of the feature data remains unchanged

- Activation functions:
  - **Sigmoid**: $\sigma(x) = \frac{1}{1+e^{-z}}$
  - **Tanh**: $tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$
  - **ReLU(Rectified Linear Unit)**: $ReLU(z) = max(0, z)$



- Neural network architecture
  - Naming convention: $N_{layer} = (N-1)$(layers of hidden units) $+1$(output layer), input layer is not counted